

**IN THE SPECIFICATION**

Please amend the specification as follows:

1. In the title:

**METHOD AND APPARATUS FOR CONTROLLING RESULT DATASET  
GENERATION IN A JAVASCRIPT JavaScript ENVIRONMENT**

2. At page 1, lines 6-13:

U.S. Patent Application No. \_\_\_\_\_, filed \_\_\_\_\_, and entitled, "Method and Apparatus for Argument Parameterization of Complex Dataset Operations"; U.S. Patent Application No. \_\_\_\_\_, filed \_\_\_\_\_, and entitled, "Method and Apparatus for Dataset Manipulation in a Javascript Environment"; U.S. Patent Application No. \_\_\_\_\_, filed \_\_\_\_\_, and entitled, "Cool ICE data Wizard"; U.S. Patent Application No. \_\_\_\_\_, filed \_\_\_\_\_, and entitled, "Cool ICE Column Profiling"; U.S. Patent Application No. \_\_\_\_\_, filed \_\_\_\_\_, and entitled, "Cool ICE OLEDB Consumer Interface"; and U.S. Patent Application No. \_\_\_\_\_, filed \_\_\_\_\_, and entitled, "Cool ICE State Management" are commonly assigned co-pending applications ~~incorporated herein by~~ reference.

3. At page 10, lines 3-15:

In the preferred implementation, the JavaScript parser and objects are integrated into the MAPPER engine to support JavaScript stored procedures. The integrated JavaScript parser interprets and executes JavaScript stored procedures, which utilize custom JavaScript objects. These custom capabilities in an object-based, paradigm for dataset manipulation and analysis purposes. Additional custom JavaScript objects are also provided to support the more complex MAPPER core engine “~~power~~” power function analysis capabilities. JavaScript stored procedures are an alternative to MAPPER run-script, input and output arguments can be passed, and a resulting dataset can be returned to the caller

A key to making this process efficient is the technique for “parameterization” of the underlying MAPPER “~~power~~” power commands. In order to leverage the more complex MAPPER core engine “power” function analysis capabilities, it is necessary for the programmer to supply a set of arguments. The arguments are positional and the number can range from just a few to many dozens. As the number of arguments increases, the burden of programming them can become unmanageable.